

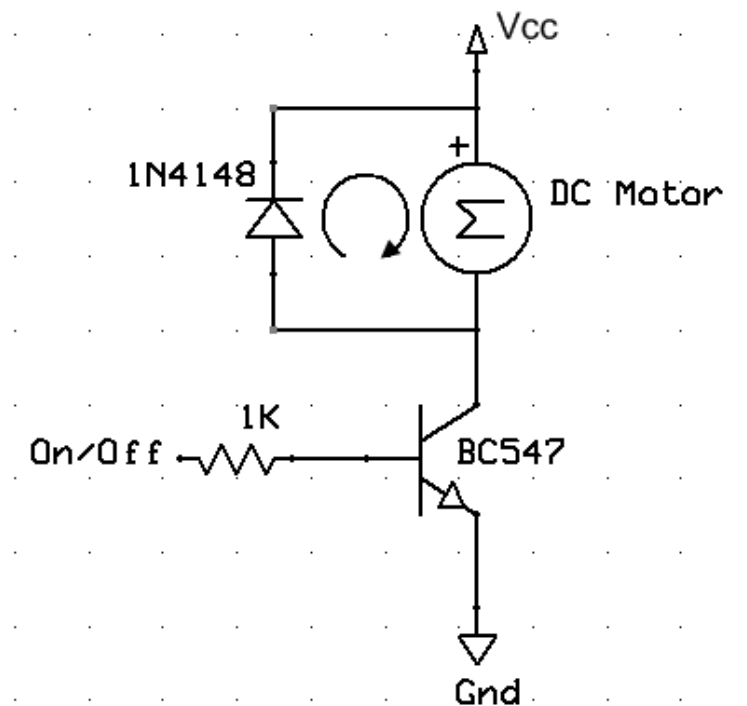
تستخدم المحركات الكهربائية ذات التيار الثابت (DC Motor) بشكل كبير في هذه الايام وخاصة في تصميم الروبوتات رغم ان البعض يفضل استخدام محركات السيرفو Servomotor ... وفي الواقع كلا النوعين لهما اهمية كبيره في تصميم الاجهزة الكهربائية لذلك في هذه المحاضرة سيكون الشرح عن التحكم بالمحركات الكهربائية في كلا النوعين سواء DC motor أو Servomotor ... وستكون محاضرة في غاية الروعه والاهمية باذن الله تعالى

التحكم بـ DC motor

نعلم جميعاً ان لدينا ما هي خصائص المحركات الكهربائية ذات التيار الثابت والاهم من ذلك هو معرفة نوع المحرك الذي سنتحكم به ومقدار التيار المسحوب ومقدار الفولتية التي يجب تطبيقها على اطرافه ... وكل هذه المعلومات تكون متوفره في الداتاشيت الخاصة بالمحرك .

لكن عادتاً تكون الفولتية المطلوبه لتشغيل المحرك لا تقل عن 7.5 V ، ولكن كيف سنحصل على هذه الفولتية ونحن نعرف بأن المايكروكنترولر لن يزودنا أكثر من 5 فولت البعض يستخدم الـ 5 فولت لتشغيل المحرك وفي بعض الاحيان قد تنجح ... ولكن للأسف يؤدي هذا لحرق المتحكم .

ان المايكروكنترولر يزودنا بـ 5V ويعطينا ما يقارب 20 mA على اطرافه ولكن في بعض الاحيان وخاصة عند تحميل المحرك بحمل ولو كان بسيطاً يؤدي هذا كله الى زيادة التيار المسحوب وهذا بتأكيد يؤدي الى حرق المايكروكنترولر لذلك الحل الافضل هنا هو استخدام ترانزستور وتشغيله كمفتاح لذلك لابد لنا من التعرف على طريقة تشغيل الترانزستور كمفتاح



أولاً لابد من حساب قيمة تيار الـ Base يلي:

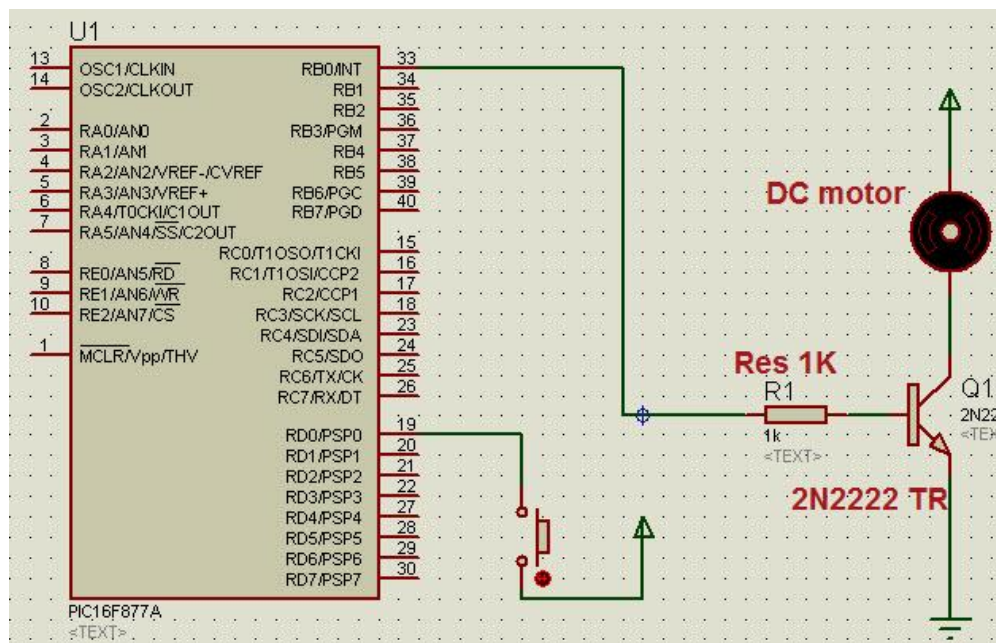
$$I_B = \frac{I_C}{\beta}$$

ومن ثم نحسب قيمة المقاومة من العلاقة التالية :

$$R_B = \frac{V_{in} - V_{BE}}{I_B}$$

وبهذا يمكن حساب قيمة المقاومة وذلك بمعرفة التيار الذي يحتاجه المحرك (بشكل عام الحمل المراد تشغيله)

والان لنأخذ مثال بسيط على تشغيل المحرك الكهربائي ونوصل الدائرة التالية



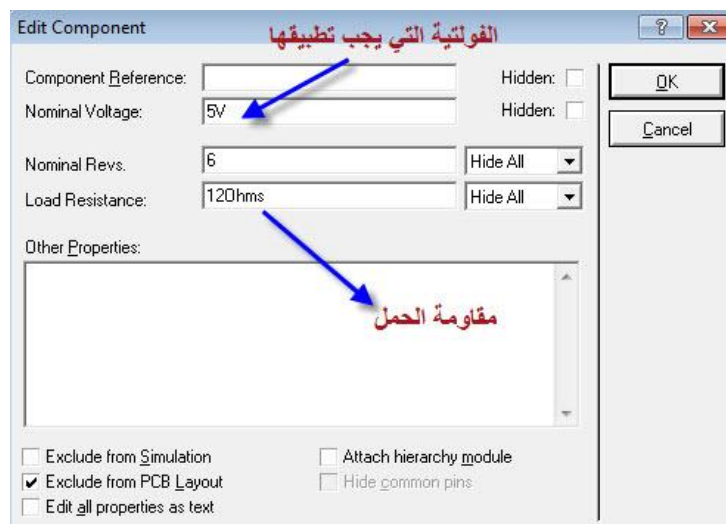
والان ان طريقة البرمجة بسيطة جداً وتستطيع انتا برمجتها فكل الفكرة هي اخراج 5V على الطرف الموصول بالترانزستور .

وهذا قد يكون احد البرامج التي تفيد عملية التحكم

```
void main() {
  trisb=0x00 ;
  trisd=0x01 ;
  portb=0x00 ;
  while(1){
    if(portd.f0==1)
    { portb=0x01 ;}
    else
    {portb=0x00 ;}
  }
}
```

فمن خلال هذا البرنامج قمنا بتجهيز B0 بأنه خرج وايضاً D0 بأنه دخل وعند النقر على الـ Button يخرج ٥ فولت وهذا يؤدي الى تشغيل الترانزستور وبالتالي يعمل المحرك .

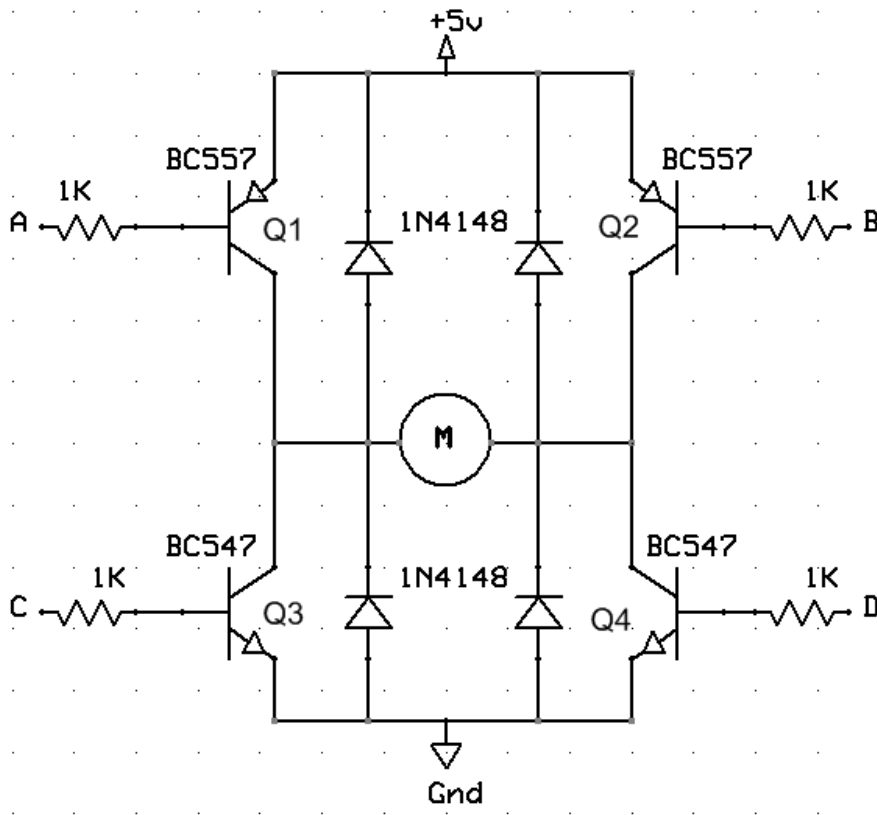
وهنا لو نظرنا نقرتين مزدوجتين على المحرك ستظهر لنا القائمة التالية التي من خلالها من الممكن التحكم بقيمة الـ Nominal Voltage للمحرك الكهربائي وهي قيمة الفولتية التي يجب توصيلها على طرفي المحرك ، كما ولاحظ وجود قيمة مقاومة الحمل التي من خلالها يمحساي تيار الحمل وبالتالي المقامة الموصله مع الترانزستور.



والآن ماذا لو اردنا التحكم بالمحرك بحيث نتمكن من عكس حركته؟؟

ان الامر في غاية السهولة فكلنا نعلم ان التحكم بحركة الـ DC Motor تتم من خلال عكس اتجاه مرور التيار خلال الملفات اذاً تكمن الفكرة بالتفكري بطريقة ما لعكس حركة التيار وبالتالي عكس اتجاه دوران المحرك

يتم ذلك من خلال استخدام دائرة خاصة تسمى بـ H-Bridge وهي ببساطة عبارة عن اربعة ترانزستور كما في الدائرة التالية

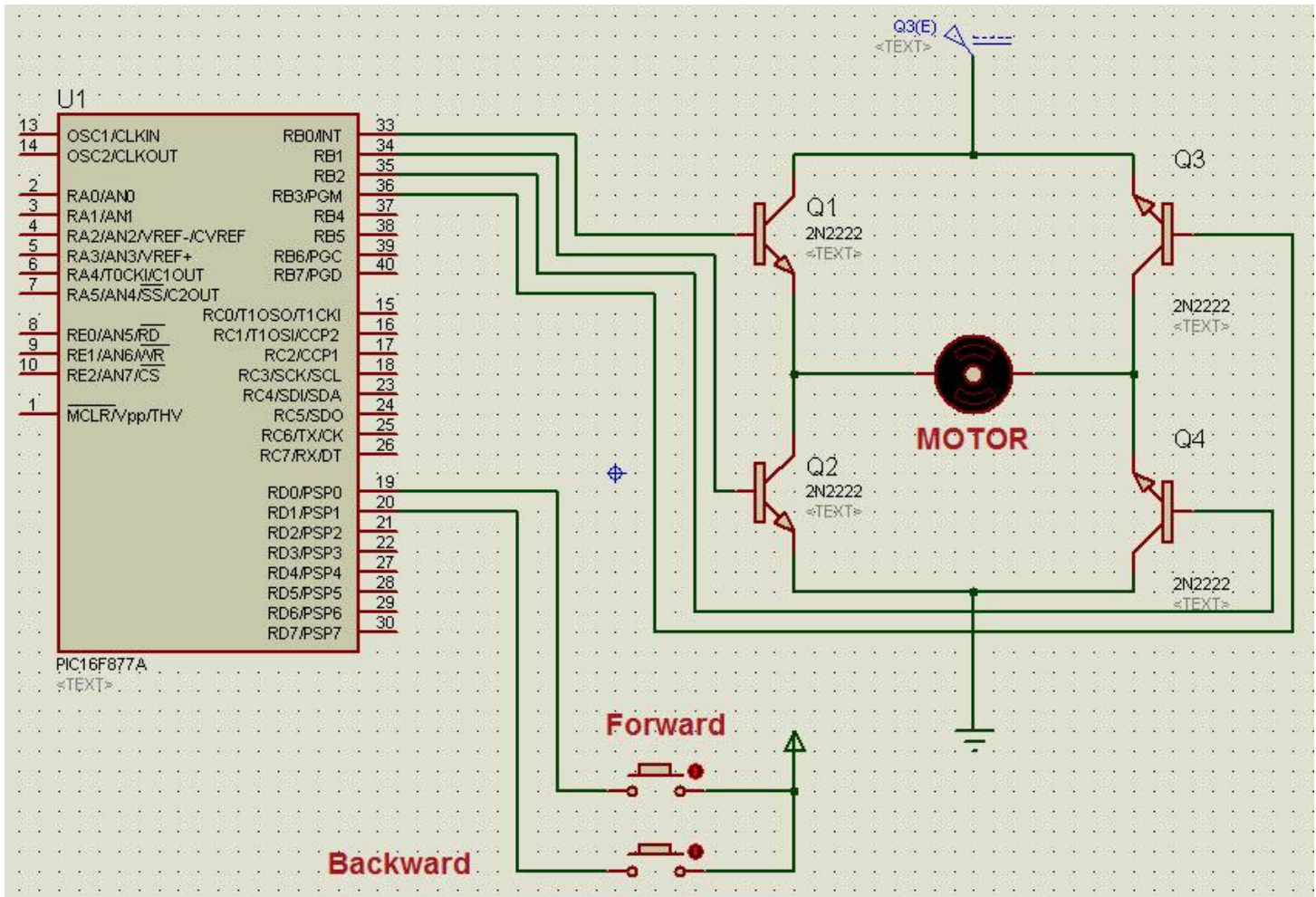


ولفهم هذه الدائرة تخيل بأننا وصلنا ه فولت على الترانزستور رقم Q1 و Q4 ... عندا يؤدي هذه الى مرور التيار من خلالهما وبالتالي فان حركة المحرك ستكون باتجاه عقارب الساعة

اما عن تشغيل الترانزستور رقم Q2 و Q3 سينعكس اتجاه حركة التيار في المحرك وبالتالي سيدور باتجاه عكس عقارب الساعة .

ويتم توصيل اطراف الترانزستور الاربعة مع المايكروكنترولر وبالتالي فان عملية التحكم ستكون في غاية من السهولة

والان لنطبق هذه الدائرة على هذه الدائرة على برنامج البروتس كما في الصورة التالية



وهنا يجب الانتباه الى ما يلي ...

لدوران المحرك مع عقارب الساعة يجب ان يكون B0 , B2 هما Logic one (٥ فولت) والبقية صفر
لدوران المحرك عكس عقارب الساعة يجب ان يكون B1,B4 هما Logic one (٥ فولت) والبقية صفر

وبالتالي بنانا على هاتين المعلومتان يمكننا كتابة البرنامج بحيث عند النقر لى الـ Forward يتحرك المحرك مع عقارب الساعة والعكس عند النقر على Backward

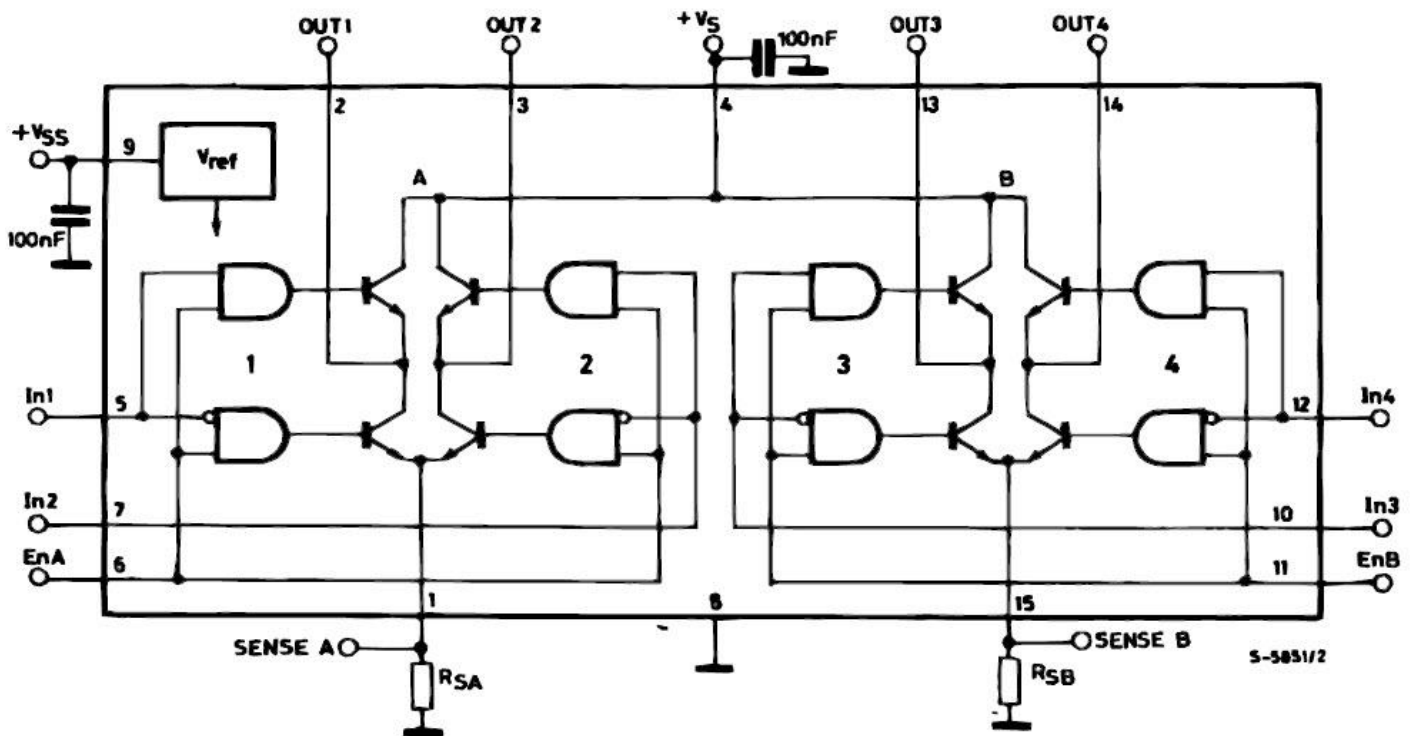
وبالتالي من الممكن ان نستخدم هذا البرنامج لتنفيذ هذا الهدف ... وبالطبع يمكن أي شخص ان يكتب برنامجاً مختلفاً
فلك الرأي في ذلك .

```

void main(){
    trisb=0x00 ;
    trisd=0xff ;
    portb=0x00 ;
    while(1) {
        if(portd.f0==1 ;)
        {portb=0b00001010 ;}
        if(portd.f1==1;)
        {portb=0b00000101 ;}
    }
}

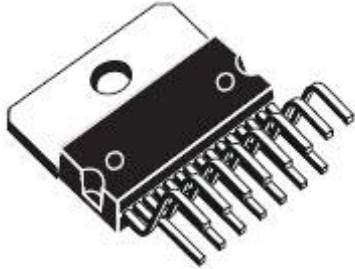
```

ولحسن الحظ هناك قطعة إلكترونية خاصة لتشغيل الـ DC motor وهي تحتوي نفس الدائرة التي طبقناها سابقاً ولكن موجوده بقطعة واحدة وهي قادره على تشغيل محركين أي بمعنى آخر انه تحتوي على دائرتين من نفس الدوائر السابقة وهي تسمى بـ H-Bridge ... والدائرة الموجوده بداخلها هي كما يلي

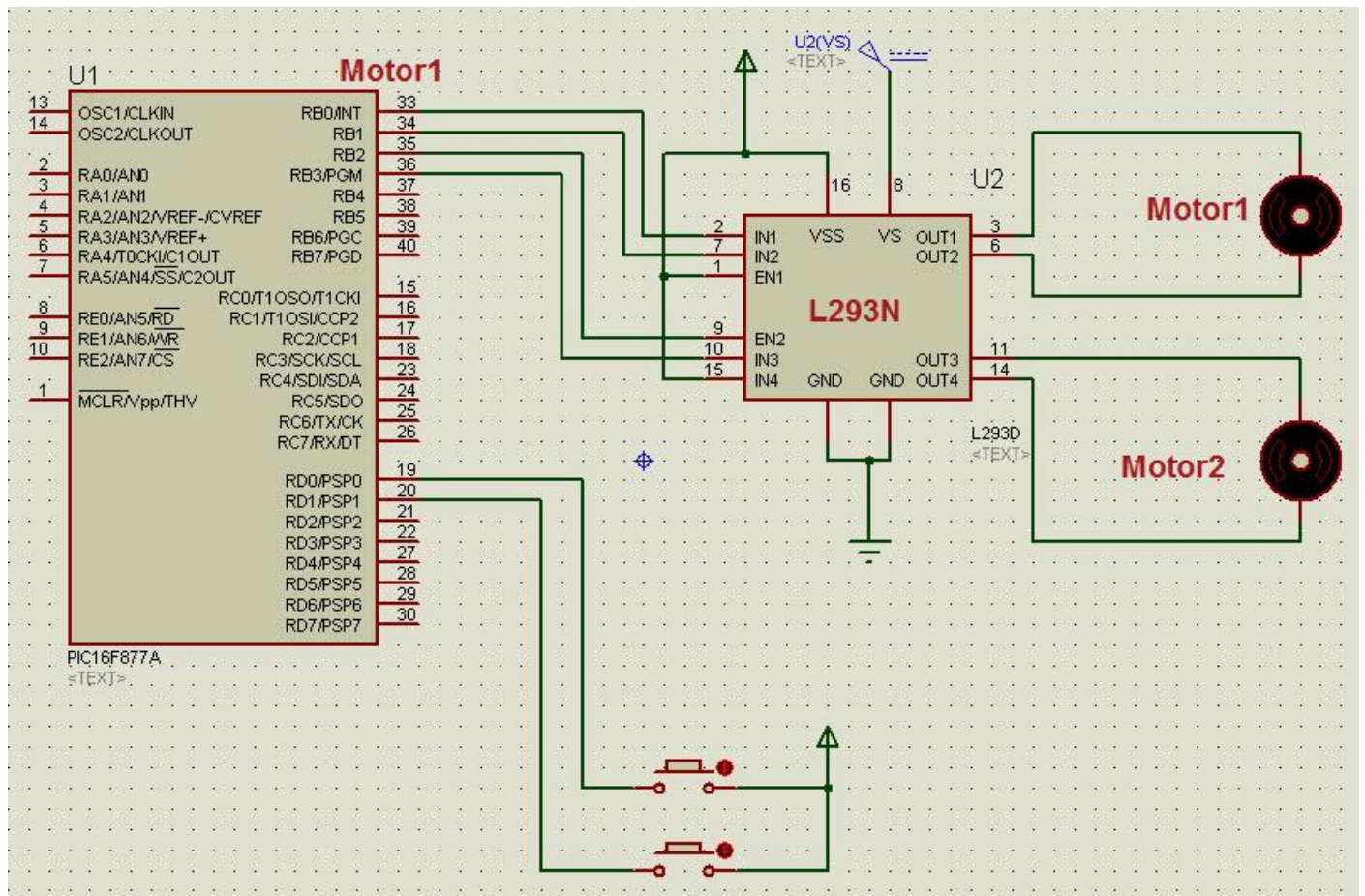


لاحظ ان على الطرفين OUT1 , OUT2 يوصل بينهما المحرك الاول ... ويتم التحكم به من خلال الطرفين In1,In2
وبتأكيد يجب توصيل الطرف ENA مع VCC ونفس الشيء بالنسبة لتوصيل المحرك الثاني

وهذه الاطراف تكون هي فقط الخارجة من القطعة كما في الصورة التالية



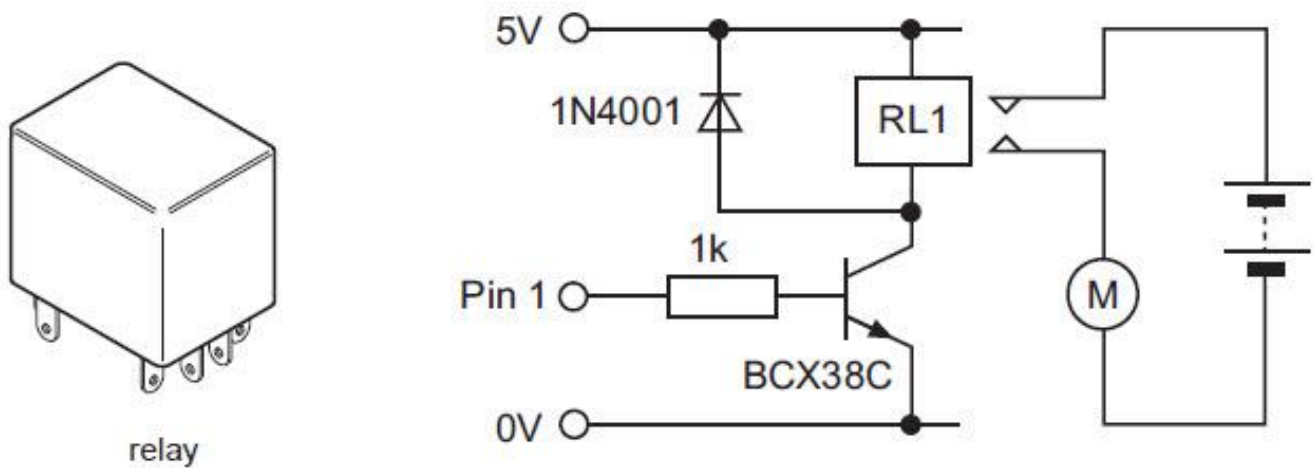
ولحسن الحظ هذه القطعة متوفرة على برنامج بروتس تحت اسم L293N وهذه دائرة تبين طريقة التوصيل كما شرحنا في الاعلى



قم ببرمجة المايكروكنترولر لتحكم في المحركين بالطريقة التي تراها مناسبة فهذا لن يختلف كثيرا عن الطريقة السابقة
سوا طريقة توصيل القطعة الجديده

وقبل الانتهاء من التحكم DC motor لابد من الاشارة هنا ان الترانزستور قد لا يتحمل فولتيات عالية ... وماذا لو اردنا التحكم بمحرك يعمل على ٢٢٠ فولت مثلاً فلا بد ان يكون هنا طريقه ما !!

نعم ف مثل هذه الحالة نستخدم ريليه فيتم توصيل المحرك مع الكنتاكت الخاص بالريليه أما الملف فيوصل مع ترانزستور وبهذا عن وجود خمسة فولت يتم تشغيل الترانزستور وبالتالي شحن الملف الخاص بالريليه ومنها اغلاق الكنتاكت وبالتالي تشغيل المحرك وهذا كما في الدائرة التالية



وبهذا نكون قد انتهينا من التحكم بـ DC Motor والان ننتقل الى الجزء الثاني من المحاضرة وهو التحكم بمحركات السيرفو Servomotor

التحكم بـ Servomotor

السيرفو موتور هو عبارة عن DC motor مجهز بدائرة الكترونيه للتحكم بدقة في اتجاه دوران عمود الموتور

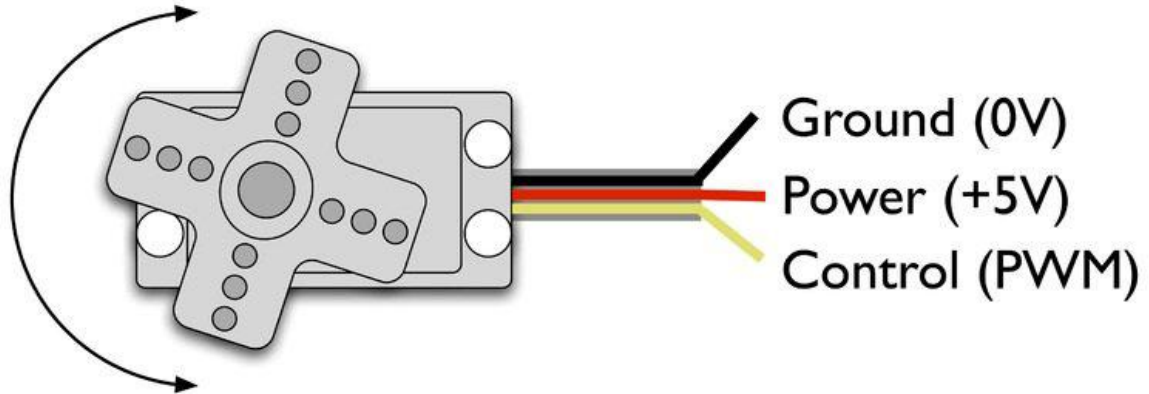
يوجد نوعين من محركات السرفو وهما:

standard servo : وهو يدور من 0 - 180 درجة

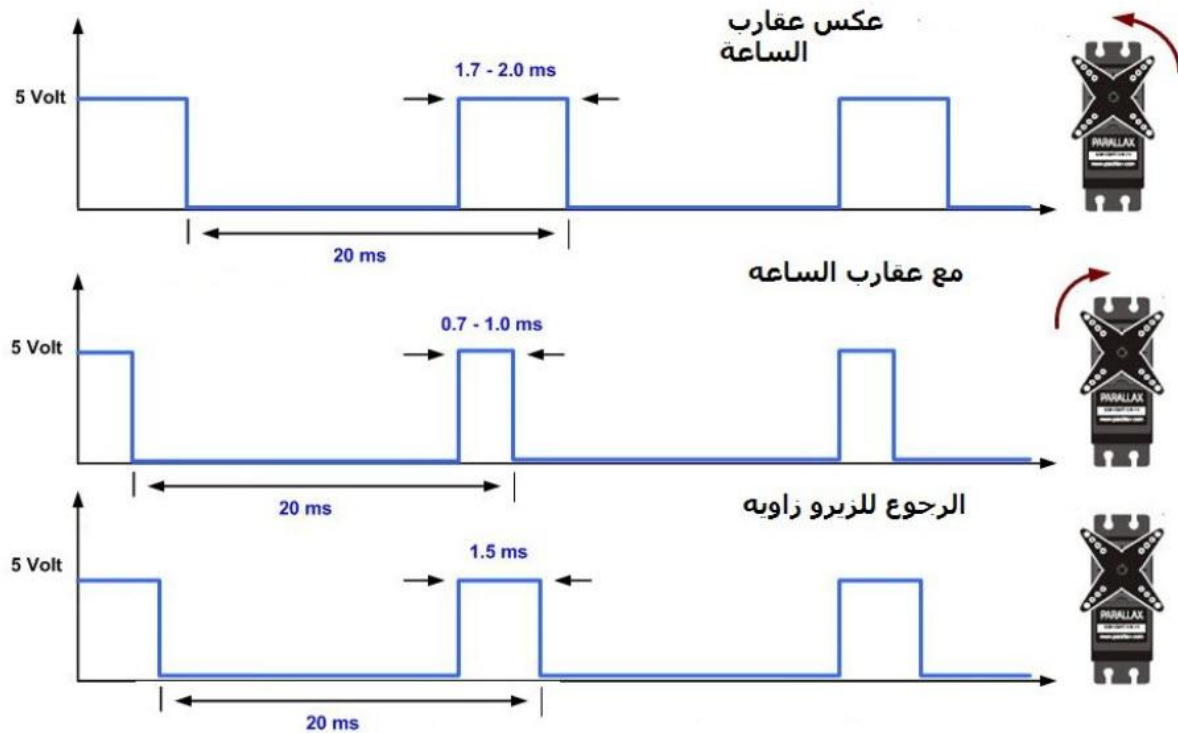
contineous servo motor : ويدور من 0 - 360 درجة

وما يهمنا الان هو طريقة التحكم فيه

كما نلاحظ في الصورة التالية يوجد ثلاثة اسلاك خارجة من الموتور اسود و احمر و اصفر
الاسود يتصل بالارضى و الاحمر بموجب 5 فولت و الاصفر اشارة التحكم و هي مجموعة نبضات بتردد 50 Hz



وبما ان التردد المطلوب هو 50 Hz اذا هذا يقابل 20 ms وبما ان النبضة عباره عن Low و high بالتالي فان مجموع كل منها يجب ان يساوي 20 ms وللحصول على زاوية دوران معينه من قبل المحرك يتم تزويده باشاره (مجموعه من النبضات) المقابله لهذه الزاوية وذلك بحسب الصورة التالية



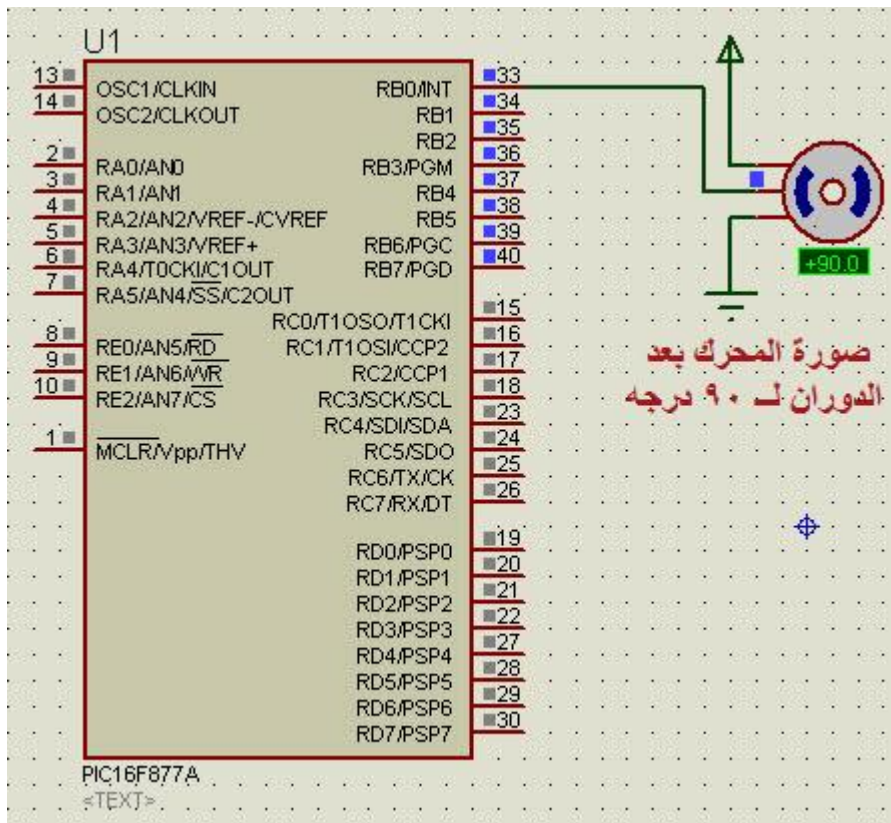
اذا لدوران عكس عقارب الساعة تكون عرض النبضة الـ high ما بين 1.7 - 2 ويجب الانتباه ان عرض النبضة كامله يجب ان تكون 20 ms لان التردد هو 50 Hz ولا بد الرجوع لداتاشيت لتأكد من التردد المطلوب .

وبالتالي يمكن ان نكتب الكود المخصص لدوران المحرك الى اقصى زاوية باتجاه عقارب الساعة كما يلي
مع الملاحظة ان بنقر المزدوج على المحرك ببرنامج المحاكاه يمكن ضبط اقصى زاويه له وهي بشكل افتراضي ٩٠ درجة .

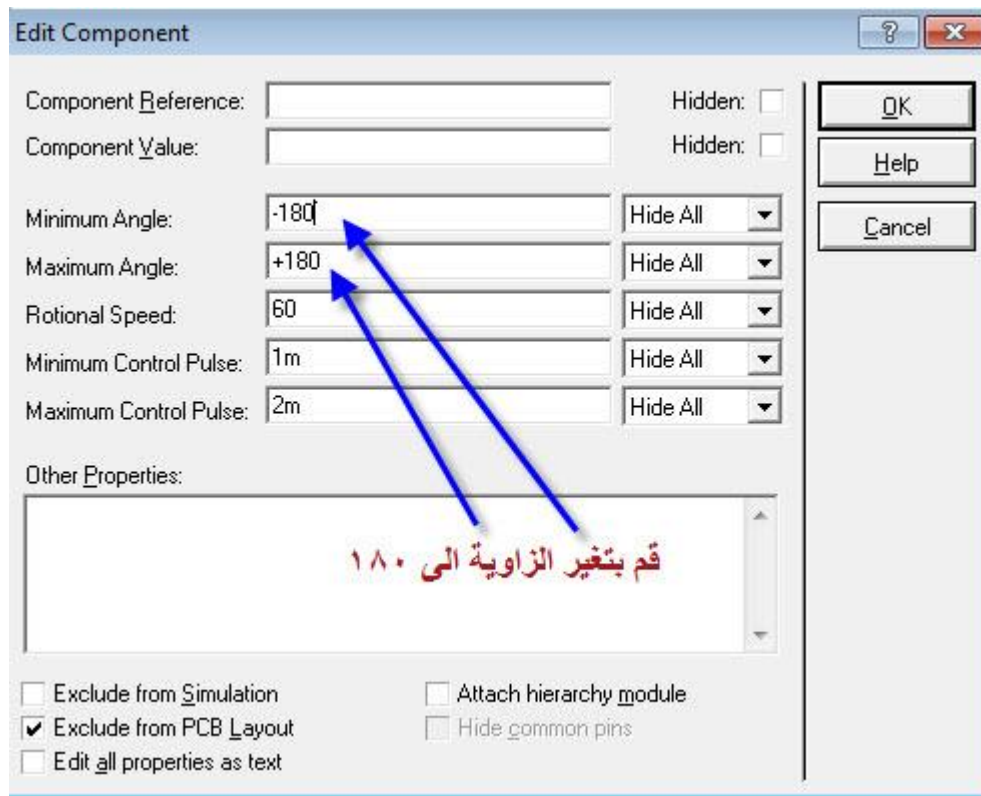
```
void main() {
  TRISB=0;
  PORTB=0;
  while(1) {
    portb.f0=1;
    delay_us(700);
    portb.f0=0;
    delay_us(18300);
  }
}
```

لاحظ ان حاصل جمع $20000 = 700 + 18300$ وهي 20 ms

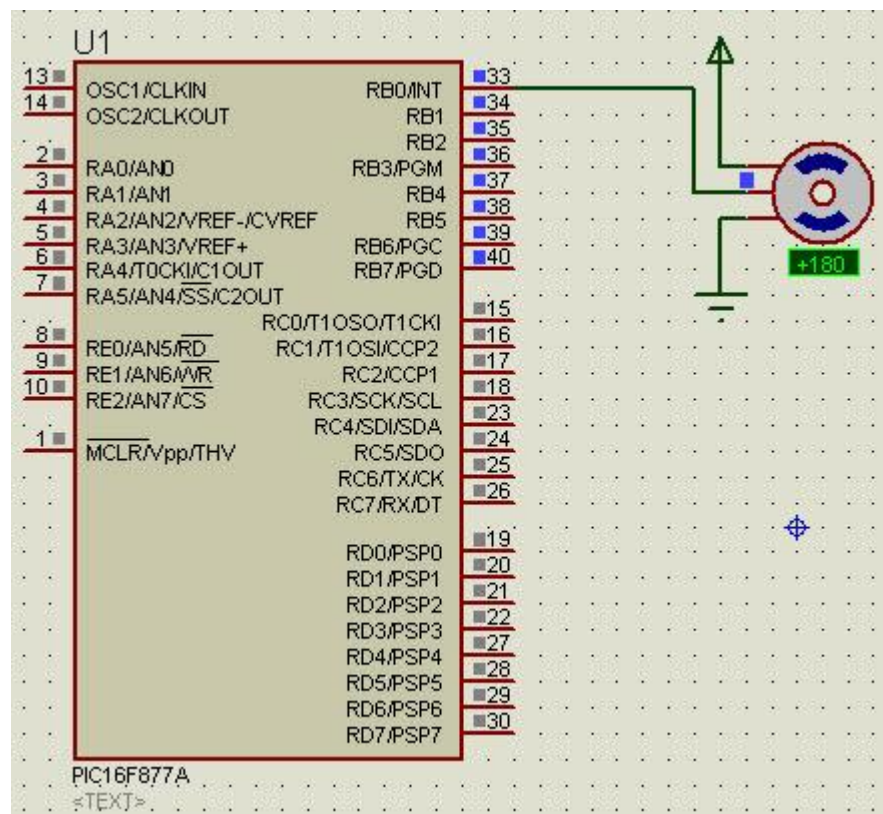
وبالنسبة لتوصيل المحرك بالمايكروكنترولر موضحة كما في الدائرة التالية



وهنا بعد النقر المزدوج على المحرك ستظهر لنا الشاشة التالية التي منها نختار قيمة الزاوية القصوى لدوران المحرك وقم بتغييرها من ٩٠ الى ١٨٠ واعد تشغيل البرنامج ولاحظ الفرق .



وبعدالزاويه و تشغيل البرنامج مره لاحظ التغير على زاوية دوران للمحرك



والى هنا ننتهي من المحاضرة الرابعه بحمد الله

بانتظار اسألتكم واستفساراتكم وملاحظاتكم